UniLDiff: Unlocking the Power of Diffusion Priors for All-in-One Image Restoration

Supplementary Material

1. Description of the Load Balancing Losses

We describe below the regularizers that we use to enforce a balanced usage of the experts.

1.1. Importance Loss.

We incentivize a balanced usage of experts via an importance loss. The importance of expert i for a batch of images $\mathbf{X}$ is defined as the normalized routing weight corresponding to expert i summed over images:

$$\text{Imp}_i(\mathbf{X}) := \sum_{\mathbf{x}\in\mathbf{X}} \text{softmax}(W\mathbf{x})_i, \tag{1}$$

where W is the layer-specific weight matrix for the router. We use the squared coefficient of variation of the importance distribution over experts, $\text{Imp}(\mathbf{X}) := \{\text{Imp}_i(\mathbf{X})\}_{i=1}^{E}$:

$$\mathcal{L}_{\text{Imp}}(\mathbf{X}) = \left(\frac{\text{std}(\text{Imp}(\mathbf{X}))}{\text{mean}(\text{Imp}(\mathbf{X}))}\right)^2 \propto \text{var}(\text{Imp}(\mathbf{X})). \tag{2}$$

1.2. Load Loss.

The importance loss seeks to guarantee that all experts have on average similar output routing weights. Unfortunately, it is not difficult to construct routing configurations where these weights are balanced overall, but a small subset of experts get all the assignments.

Ideally, we would also like to explicitly balance the number of assignments. This quantity is discrete; therefore it is not differentiable, and we need to rely on a proxy. For each token $\mathbf{x}$, we compute the probability of expert i being selected — i.e., being among the top-k — if we were to re-sample only the noise for expert i. For each token $\mathbf{x}$, we define the score threshold above which experts were selected:

$$\text{threshold}_k(\mathbf{x}) := \max_{k\text{-th}}(W\mathbf{x} + \epsilon), \tag{3}$$

where ϵ was the noise vector originally sampled during the forward pass. Then, for each expert i we compute the probability of i being above the threshold if we were to only re-sample its noise:

$$\begin{aligned} p_i(\mathbf{x}) &:= \mathbb{P}\left((W\mathbf{x})_i + \epsilon_{\text{new}} \geq \text{threshold}_k(\mathbf{x})\right) \\ &= \mathbb{P}\left(\epsilon_{\text{new}} \geq \text{threshold}_k(\mathbf{x}) - (W\mathbf{x})_i\right) \end{aligned} \tag{4}$$

The probability is defined over $\epsilon_{\text{new}} \sim \mathcal{N}(0, \sigma^2)$, with $\sigma = 1/E$. The load for expert i over batch $\mathbf{X}$ is:

$$\text{load}_i(\mathbf{X}) = \sum_{\mathbf{x}\in\mathbf{X}} p_i(\mathbf{x}). \tag{5}$$

Finally, the load loss corresponds to the squared coefficient of variation of the load distribution:

$$\mathcal{L}_{\text{load}}(\mathbf{X}) = \left(\frac{\text{std}(\text{load}(\mathbf{X}))}{\text{mean}(\text{load}(\mathbf{X}))}\right)^2 \tag{6}$$

where $\text{load}(\mathbf{X}) := \{\text{load}_i(\mathbf{X})\}_{i=1}^{E}$.

1.3. Final Auxiliary Loss.

The final auxiliary loss is just the average over both:

$$\mathcal{L}_{\text{aux}}(\mathbf{X}) = \frac{1}{2}\mathcal{L}_{\text{Imp}}(\mathbf{X}) + \frac{1}{2}\mathcal{L}_{\text{load}}(\mathbf{X}). \tag{7}$$

The overall loss is: $\mathcal{L}(\mathbf{X}) = \mathcal{L}_{\text{classification}}(\mathbf{X}) + \lambda\mathcal{L}_{\text{aux}}(\mathbf{X})$, for some hyperparameter $\lambda > 0$. We set $\lambda = 0.01$ in all our experiments, observing that this choice was robust and not sensitive.

2. Experiment

2.1. Degradation Datasets

Three Degradation Setting. For both the All-in-One and single-task settings, we follow the standard evaluation protocols established in prior works, and utilize the following datasets. For image denoising in the single-task setting, we combine the BSD400 and WED datasets and corrupt the images with Gaussian noise at levels $\sigma \in \{15, 25, 50\}$. BSD400 contains 400 training images, while WED includes 4,744 training images. We evaluate denoising performance on BSD68 and Urban100. For single-task deraining, we use the Rain100L dataset, which provides 200 clean/rainy image pairs for training and 100 pairs for testing. For single-task dehazing, we adopt the SOTS dataset, consisting of 72,135 training images and 500 testing images. Under the All-in-One setting, we train a unified model on the combined set of the aforementioned training datasets and directly test it across all three restoration tasks.

Five Degradation Setting. The 5-degradation setting is built upon the 3-degradation setting by including two additional tasks: deblurring and low-light enhancement. For deblurring, we adopt the GoPro dataset, which contains 2,103 training images and 1,111 testing images. For low-light enhancement, we use the LOL-v1 dataset, consisting of 485 training images and 15 testing images. Note that for the denoising task under the 5-degradation setting, we report results using Gaussian noise with $\sigma = 25$.

Composite Degradation Setting. For the composite degradation setting, we use the CDD11 dataset. CDD11 consists of 1,183 training images that cover: (i) four kinds of

single-degradation types: haze (H), low-light (L), rain (R), and snow (S); (ii) five kinds of double-degradation types: low-light + haze (L+H), low-light + rain (L+R), low-light + snow (L+S), haze + rain (H+R), and haze + snow (H+S); and (iii) two kinds of triple-degradation types: low-light + haze + rain (L+H+R) and low-light + haze + snow (L+H+S).

Zero-shot Under-display Camera Restoration. For the zero-shot under-display camera restoration task, we use a dataset collected through a Monitor-Camera Imaging System (MCIS). The dataset consists of paired display-free and display-covered imaging data, captured under two types of displays: T-OLED and P-OLED. A total of 300 images from the DIV2K dataset were selected and adapted to create this new dataset.

The images in the dataset have a resolution of 1024x2048 and are provided in both 16-bit RAW sensor data and 8-bit RGB formats. The dataset includes images captured under two conditions: display-free (when the display is not active) and display-covered (when the display is covering the camera sensor).

For training purposes, the dataset includes the paired RGB data. The RGB images are linear, which allows for the reversal of the process to generate 8-bit RAW sensor data. This dataset is designed for restoring images captured by under-display cameras, where part of the camera's sensor is obstructed by the display.

2.2. Implementation Details

We adopt the base model of SDXL as our latent diffusion backbone and employ its VAE encoder as the local quality encoder. All experiments are conducted on two NVIDIA A800 GPUs using the AdamW optimizer with default hyperparameters. During training, images are randomly cropped into patches of size 512×512, and the batch size is set to 64.

We first pre-train the proposed feature alignment module with a learning rate of 5×10^{-5} for 6,000 iterations. Then, the entire network, including the local quality encoder, DAFF, and the diffusion model, is jointly fine-tuned for 40,000 iterations. In this stage, the learning rates for the LQ encoder and other components are initialized to 5×10^{-6} and 1×10^{-5}, respectively. A cosine annealing schedule is used to gradually decay the learning rates.Afterward, we freeze all components except the VAE decoder, which is further fine-tuned for 10,000 iterations to refine reconstruction quality.

To enhance controllability, we adopt classifier-free guidance (CFG) by randomly dropping image descriptions during training. Our experiments show that a dropout rate of 20% yields the best performance. For inference, we utilize an Euler scheduler with 20 sampling steps and set the CFG scale to 5.

2.3. More results

Additional Quantitative Results on PSNR and SSIM. To provide a more comprehensive evaluation of our model's restoration fidelity, we report full-reference results on PSNR and SSIM across five representative restoration tasks: dehazing, deraining, denoising ($\sigma = 25$), deblurring, and low-light enhancement. As presented in Table 1, our method achieves the highest PSNR in three out of five tasks and the best SSIM in two. Compared to existing non-diffusion and diffusion-based baselines, our model consistently demonstrates strong performance across both pixel-level and structural similarity metrics, further validating its effectiveness in high-fidelity image restoration.

Complete Evaluation on CDD11 Composite Degradation. The CDD11 benchmark comprises 11 degradation scenarios, including individual, two-fold, and three-fold combinations of low-light (L), haze (H), rain (R), and snow (S). In the main paper, only a subset of these results was shown due to space constraints. Here, we present the complete evaluation in Table 2. Our method achieves the highest average PSNR (29.35 dB) and SSIM (0.886) across all tasks. Notably, it outperforms prior state-of-the-art methods such as MoCE-IR-S and OneRestore on the most challenging three-degradation compositions, demonstrating its superior generalization and robustness in handling diverse and complex degradation combinations.

Experiments under AutoDIR-Style Task Settings. AutoDIR is a diffusion-based method that introduces text-prompt guidance to tackle All-in-One Image Restoration. While it is a significant baseline, its source code is not publicly released. Therefore, to fairly benchmark against AutoDIR, we follow the same six-task setting described in their paper, including dehazing, deraining, denoising, deblurring, low-light enhancement, and deraindrop restoration. We reproduce the evaluation using our method and compare against the AutoDIR-reported values. As shown in Table 3, our model consistently surpasses AutoDIR in all six tasks, achieving notably higher PSNR and SSIM across the board. While our method also incorporates textual prompts, they are used as complementary signals rather than the main source of restoration guidance. This comparison demonstrates that our framework surpasses AutoDIR under its own task setup, highlighting the robustness of our design beyond prompt-based control.

Visual Results. To further validate the effectiveness of our method, we present additional qualitative comparisons on five representative degradation types, including deraining, denoising, dehazing, low-light enhancement, and deblurring. As shown in Figure 1, our method consistently restores image details and natural textures across different scenarios, achieving visual quality close to or even better than the ground truth. We also provide qualitative results on the CDD11 dataset, which includes 11 types of combined

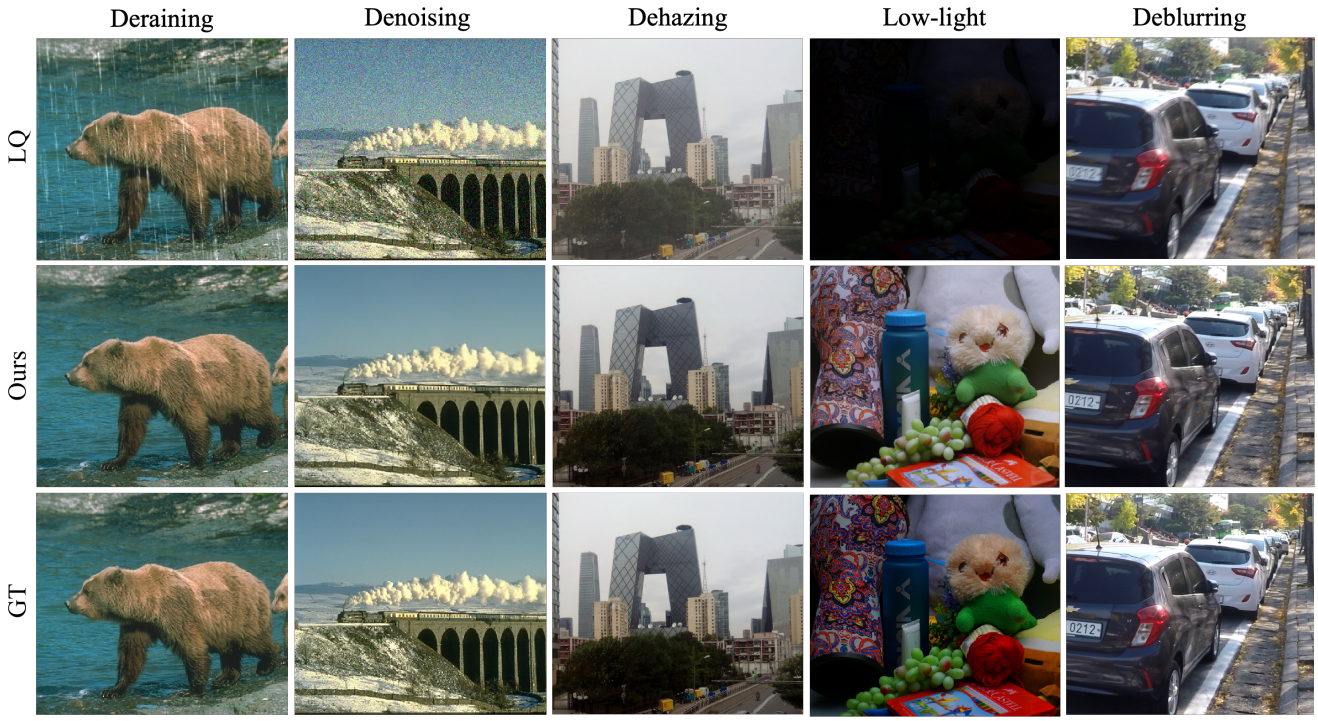

Figure 1. Qualitative comparison on five single degradation types: deraining, denoising, dehazing, low-light enhancement, and deblurring.

	Method	Dehazing		Deraining		Denoising		Deblurring		Low-light	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Non-Diff	PromptIR	29.52	0.9730	36.66	0.9757	30.90	0.8742	28.71	**0.8812**	22.46	0.8339
	InstructIR	24.95	0.8232	35.78	0.9706	31.35	0.8868	**29.55**	**0.8897**	22.79	0.8355
	AdaIR	30.25	0.9765	**37.86**	**0.9804**	31.31	0.8845	27.03	0.8523	22.94	**0.8458**
	VLU-Net	30.58	**0.9790**	**38.34**	**0.9818**	**31.40**	**0.8870**	27.48	0.8464	22.33	0.8343
	DFPIR	**31.51**	**0.9791**	37.61	0.9788	31.26	0.8845	**28.80**	0.8770	**23.66**	0.8444
Diff	DA-CLIP	28.69	0.9567	35.94	0.9701	28.98	0.8298	25.83	0.8318	21.10	0.8393
	DiffUIR	28.97	0.9297	36.47	0.9757	31.05	0.8768	26.40	0.8280	20.19	0.8319
	Ours	**30.83**	0.9583	37.80	0.9797	**31.45**	**0.8895**	27.54	0.8466	**23.74**	**0.8789**

Table 1. Comparison of PSNR / SSIM across five tasks. Bold indicates the best.

degradations, such as low-light + rain, haze + rain, and low-light + haze + snow, to evaluate model robustness under complex degradation conditions. As shown in Figure 2, our method can consistently restore image structure and natural details under different degradation combinations, achieving visual quality close to or even better than the ground truth, demonstrating strong generalization capability.

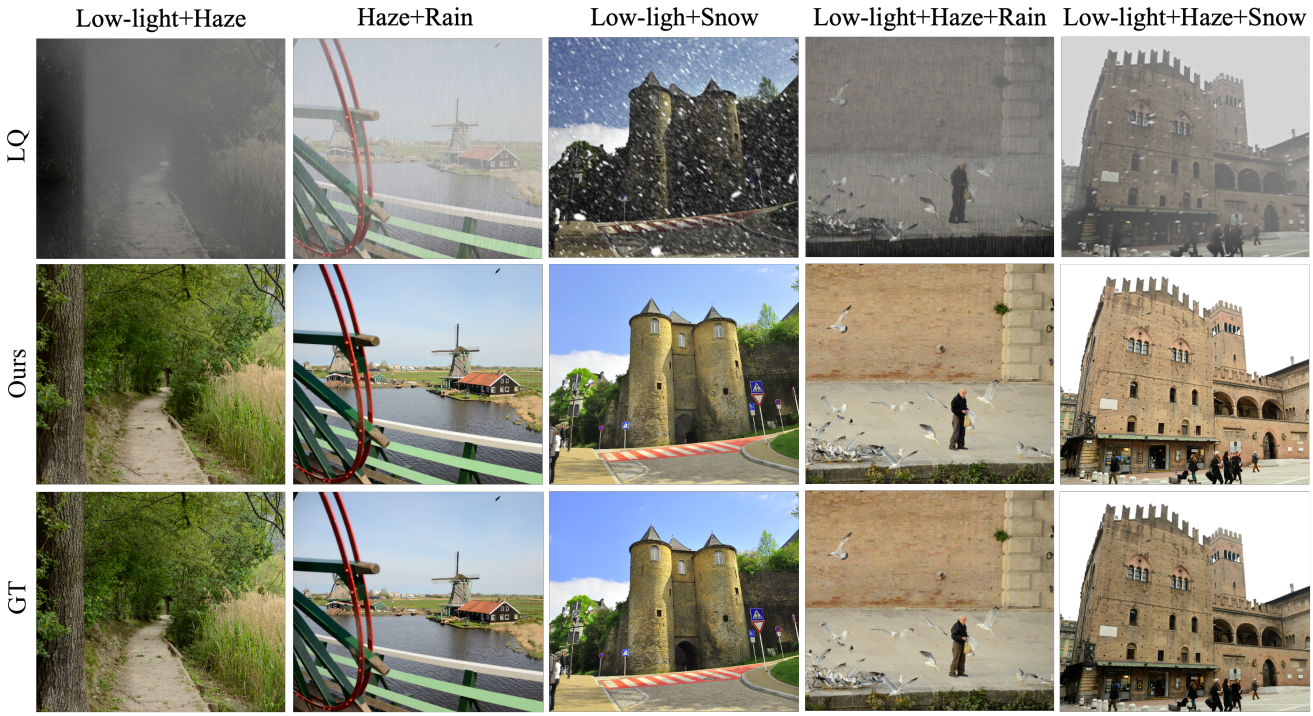

Figure 2. Qualitative comparison on the CDD11 dataset. Our method shows strong generalization and high-quality restoration under complex combined degradations.

Method	L		H		R		S		L+H		L+R		L+S		H+R		H+S		L+H+R		L+H+S		Avg.	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM		
AirNet	24.83	.778	24.21	.951	26.55	.891	26.79	.919	23.23	.779	23.21	.768	23.06	.763	25.04	.883	25.07	.884	22.85	.751	22.61	.748	23.75	.814
PromptIR	26.32	.805	26.10	.969	31.56	.946	31.53	.960	24.49	.789	24.33	.773	24.12	.770	27.93	.939	28.02	.940	23.87	.763	23.55	.759	25.90	.850
WGWSNet	24.39	.774	27.90	**.983**	31.35	.906	31.25	.906	23.95	.772	23.70	.764	23.54	.759	27.53	.890	27.60	.891	23.30	.748	23.07	.745	26.96	.863
WeatherDiff	23.58	.763	21.99	.943	24.85	.885	24.80	.883	22.36	.756	22.25	.750	22.10	.745	23.82	.867	23.91	.869	21.87	.732	21.60	.730	22.49	.799
OneRestore	26.48	**.826**	32.52	**.990**	**34.96**	.964	34.31	.973	25.79	**.822**	25.65	**.811**	25.44	**.808**	29.81	.960	29.92	.961	25.23	**.797**	24.93	**.793**	28.47	.878
MoCE-IR-S	**27.26**	.824	**32.66**	**.990**	34.31	**.970**	**35.91**	**.980**	**26.24**	.817	**26.05**	.804	**26.04**	.802	**29.93**	**.961**	**30.19**	**.962**	**25.41**	.789	**25.39**	.787	**29.05**	**.881**
Ours	**27.45**	**.830**	**32.87**	.981	**34.60**	**.972**	**36.10**	**.983**	**26.50**	**.820**	**26.40**	**.808**	**26.10**	**.805**	**30.10**	**.963**	**30.50**	**.964**	**25.80**	**.794**	**25.50**	**.790**	**29.35**	**.886**

Table 2. Comparison to state-of-the-art on 11 degradation types and their average. PSNR (dB, ↑) and SSIM (↑) are reported.

	Method	Dehazing		Deraining		Denoising		Deblurring		Low-light		Deraindrop	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Non-Diff	NAFNet	27.75	0.968	30.46	0.926	28.10	0.805	26.67	0.805	21.15	0.827	25.04	0.872
	LD	23.49	0.763	23.21	0.651	22.58	0.625	22.53	0.695	18.97	0.770	24.84	0.738
	AirNet	26.52	0.944	30.99	0.929	29.10	0.803	26.50	0.860	21.26	0.818	27.13	0.892
	PromptIR	29.13	0.971	33.97	0.938	29.89	0.824	26.82	0.819	22.42	0.831	27.41	0.900
Diff	AutoDIR	**29.34**	**0.973**	**35.09**	**0.965**	**29.68**	**0.832**	**27.07**	**0.828**	**22.37**	**0.888**	**30.10**	**0.924**
	Ours	**29.80**	**0.975**	**36.30**	**0.968**	**30.05**	**0.846**	**27.25**	**0.831**	**22.83**	**0.862**	**30.45**	**0.937**

Table 3. Comparison of PSNR / SSIM across six image restoration tasks.

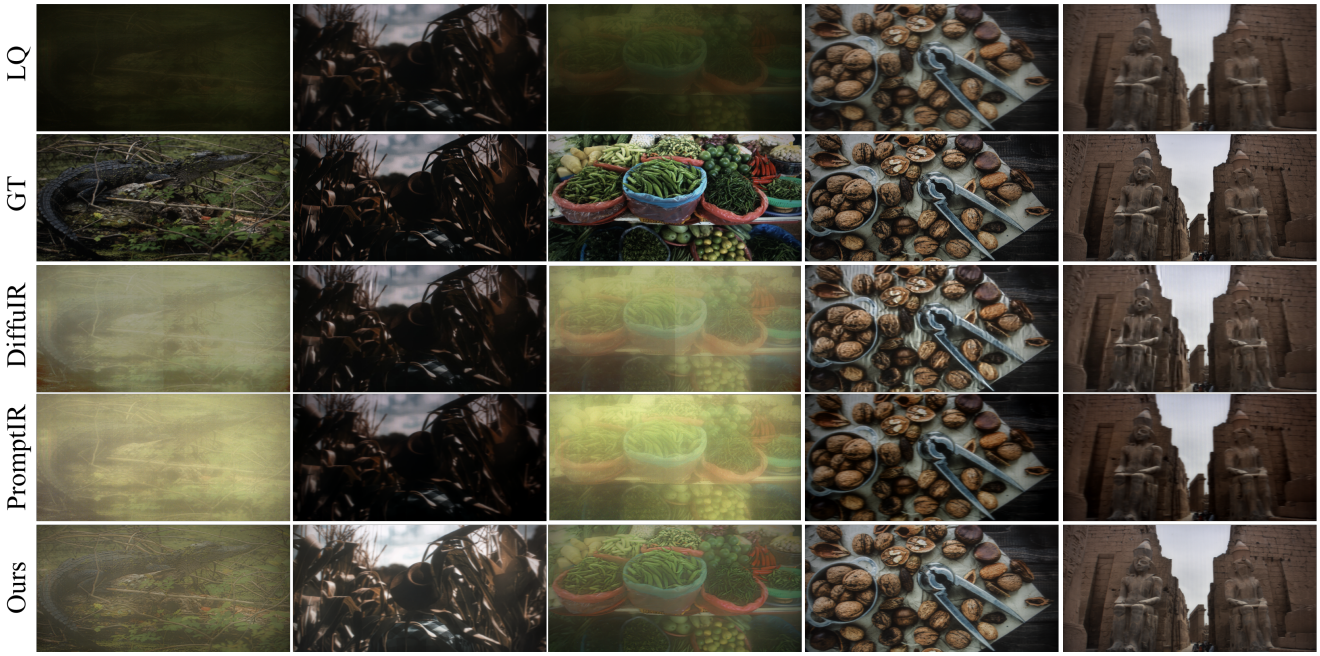

Figure 3. Qualitative comparison on the T-OLED and P-OLED dataset.